

Managed Development Environment Successes for MSFC's VIPA Team

Jeff Finckenor/NASA/MSFC/EV32

Gary Corder/Jacobs-Sverdrup

James Owens/Qualis

Jim Meehan/NASA/MSFC/EV32

Paul H. Tidwell II, Ph.D. /Allied Aerospace

Abstract

This paper outlines the best practices of the Vehicle Design Team for VIPA. The functions of the VIPA Vehicle Design (VVD) discipline team are to maintain the controlled reference geometry and provide linked, simplified geometry for each of the other discipline analyses.

The core of the VVD work, and the approach for VVD's first task of controlling the reference geometry, involves systems engineering, top-down, layout-based CAD modeling within a Product Data Manager (PDM) development environment. The top-down approach allows for simple control of very large, integrated assemblies and greatly enhances the ability to generate trade configurations and reuse data. The second VVD task, model simplification for analysis, is handled within the managed environment through application of the master model concept. In this approach, there is a single controlling, or master, product definition dataset. Connected to this master model are reference datasets with live geometric and expression links. The referenced models can be for drawings, manufacturing, visualization, embedded analysis, or analysis simplification. A discussion of web based interaction, including visualization, between the design and other disciplines is included.

Demonstrated examples are cited, including the Space Launch Initiative development cycle, the Saturn V systems integration and verification cycle, an Orbital Space Plane study, and NASA Exploration Office studies of Shuttle derived and clean sheet launch vehicles. The VIPA Team has brought an immense amount of detailed data to bear on program issues. A central piece of that success has been the Managed Development Environment and the VVD Team approach to modeling.

What is VIPA? (Reference 1)

VIPA Team History (Figure 1)

VIPA is the Vehicle Integrated Performance Analysis Team at MSFC. This team was created to reconnect the individual engineering disciplines to be able to perform system level technical assessments in support of program decisions. Early in the Space Launch Initiative (SLI) program the VIPA team was formed to provide technical insight and make NASA a "smart buyer". VIPA supported the SLI program with its first Vehicle

Analysis Cycle (VAC-00) by providing independent technical insight and supporting technical reviews. As SLI transitioned to the Orbital Space Plane (OSP), VIPA took the opportunity to prepare for upcoming needs. VIPA focused even more strongly on system interactions while validating its process against historical Saturn V data. VIPA then provided substantial performance data for evaluating the feasibility of 3 types of spacecraft design on 2 different commercial launchers. The challenge on this cycle was that 10 different configurations had to be evaluated (including the baseline and 1 configuration with two mounting schemes). After a hiatus, VIPA was called upon to support Exploration activities by performing conceptual design and analysis of potential heavy lift launchers including a shuttle derived carrier and a clean sheet design. More recent activities promise to prove VIPA's abilities for in-space systems as well as the Earth-to-orbit work already done.

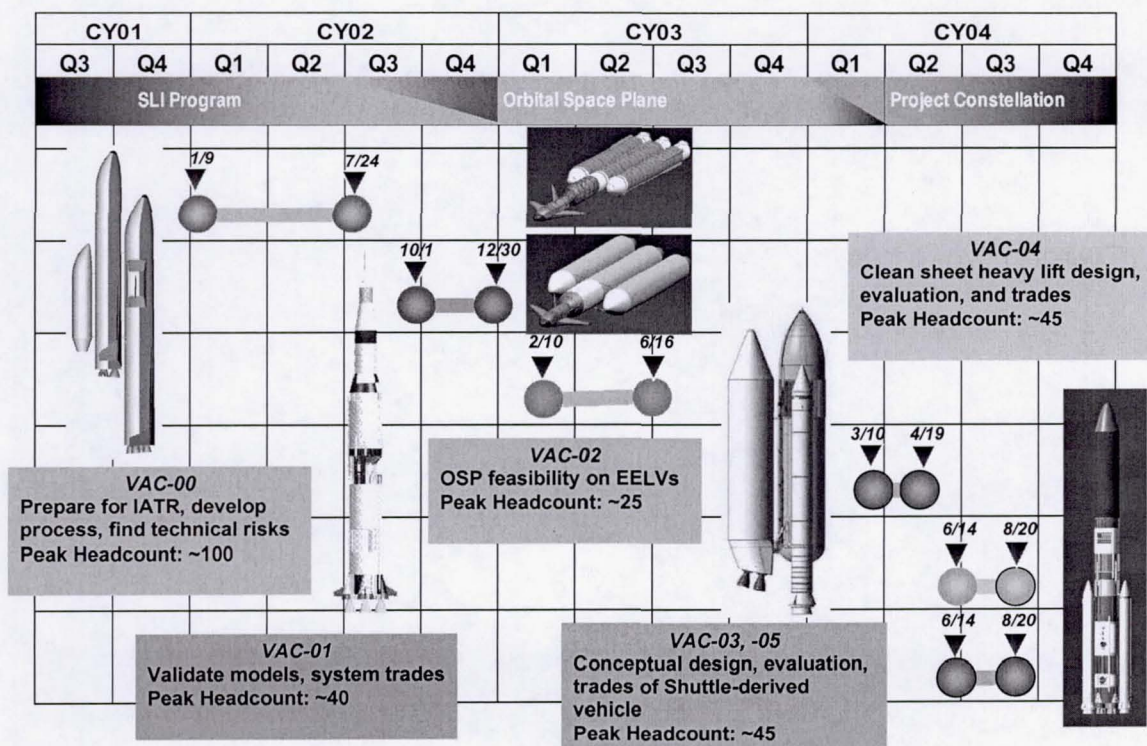


Figure 1: VIPA History

VIPA Role and Approach

“The Launch Vehicle Design Process” (Reference 2) was heavily relied upon as the team was established. The concepts of Systems Engineering and discipline and model interaction provided the foundation for the capabilities of the team.

The role of the VIPA team can be understood by examining the T-model of systems engineering (Figure 2). The legs of the model are the individual engineering disciplines, which have a long history and are well defined in their functions and methods. The top of the block is the formal control of systems engineering including requirements, resource management and project integration. These functions also have a long history

and are well defined. The lower part of the block, which can be called Analytical Integration, is an area in which processes are not well defined and is typically performed in an ad-hoc method by the individual discipline engineers involved in the project. VIPA adds rigor to this area by having the discipline leads focus heavily on the system interactions. VIPA penetration down each leg is driven only as deep as it needs to go for the system analysis. Full, detailed discipline analysis is still performed, but only when needed to support the system team.

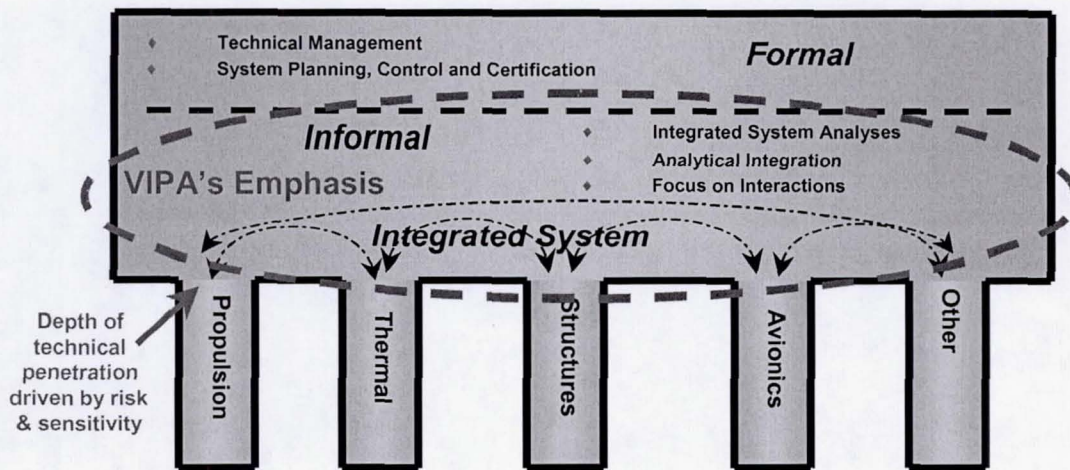


Figure 2: T-Model of Systems Engineering

The modeling and analytical approach of VIPA is illustrated in Figure 3. Three distinct types of models are defined in the "Launch Vehicle Design Process". Specialized models are very specific, often multidisciplinary analyses such as POGO (vehicle/engine dynamic interaction) or fracture mechanics. VIPA has not yet been developed in this area. Discipline-specific models are the familiar types of analysis including structural, thermal, trajectory, etc. The General model is the backbone of the systems interactions and in the case of the Earth-To-Orbit studies VIPA used MAVERIC (Marshall Aerospace VEHICLE Representation In C). MAVERIC is an MSFC developed code initiated during X-33, and is a 6 degree-of-freedom time accurate flight simulation. The modules it contains are typically simplified discipline analyses that are anchored against higher fidelity models. For VIPA, MAVERIC has demonstrated integrated simulation capabilities for: mass properties, propulsion, trajectory, guidance, navigation and control, natural environments, aerothermodynamics, thermal, loads and dynamics (including slosh), stress, avionics, dispersions and Monte Carlo analysis.

The triangle in the back of Figure 3 represents the approach for technical insight into a design originating outside of VIPA. The first step after receiving the data (the top left point) is to "Verify the Stated Performance". In this step the VIPA team makes sure that it can reproduce similar results in order to be comfortable that it understands the top-level system data and how it can be used. For example, VIPA shows that its vehicle model can fly the supplied trajectory and deliver the same payload. Then the team follows the systems engineering approach of de-integrating the design and "Validating Input Data" (bottom point). At this point the discipline teams dig into their specialty areas to make sure nothing has been missed and that there are no incorrect assumptions. An example

might be finding that all the residuals in the feedlines were not included in the burnout mass properties. Lastly, the issues found by the disciplines are “Re-Integrated” (the top right point) and their system effects determined. Accounting for the residual propellant mass significantly reduces the payload delivery for the given trajectory, however the trajectory can be re-optimized to gain back much of the lost payload.

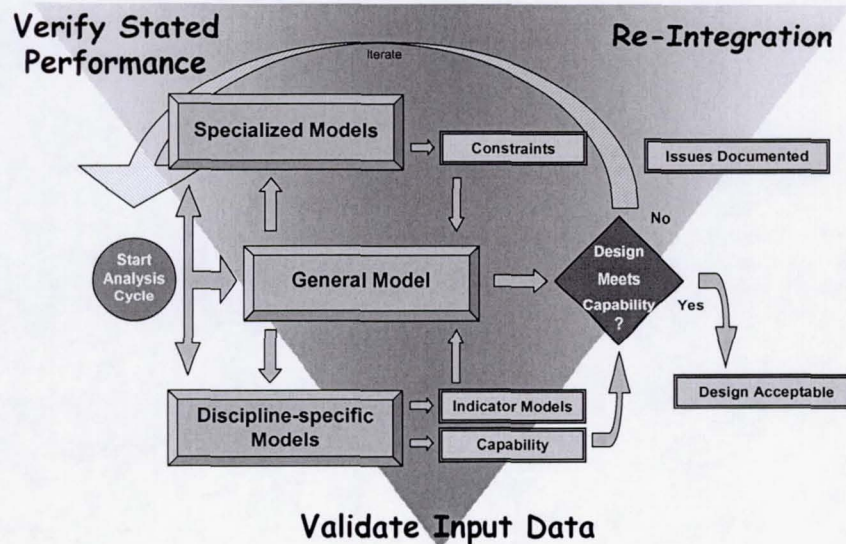


Figure 3: VIPA Approach

It is important to note that VIPA is not a tool. VIPA emphasizes the Process, the People and the Tools. While the General model is a collection of integrated analyses, the actual tool developed is not critical. What is critical is the insight that occurs between the members of the team while the tool is developed. Having the people exercise the process allows them to gain insight while using their familiar discipline models and generating the general system model. This is significantly different from analytical integration projects that often seem to focus on tying together analytical code, and result in an overspecialized and inflexible tool which cannot be easily or quickly applied to a significantly different problem.

VIPA Team Organization

The VIPA team consists of a Team Lead, the Systems Team, and the Discipline Teams. The VIPA Team Lead acts as the connection between the team and the project customer. He receives overall direction from the customer and coordinates the team direction, helping define the baseline configuration and what trades must be performed. The Discipline teams consist of Avionics, Flight Mechanics, Flight Sciences, Loads & Dynamics, Material & Manufacturing, Propulsion, Structural Assessment, Systems Modeling, Thermal, and Vehicle Design. The Systems Team consists of the lead from each discipline, the VIPA Team Lead, and Discipline team members as needed. It is the interactions among the Systems team that bring out the true insights and understanding of the system and its interactions. The VIPA Lead challenges the Discipline Leads to focus on the system more than their own discipline. They need to understand how the data they

need is being generated to make sure it is consistent with their applications. They must also know how the data they have generated is being used for the same reason.

The VIPA Vehicle Design Team Process

Top-Down, Layout-Based Design

The most important element of a robust vehicle design process is the ability to control and adjust the interfaces between design elements. By instituting a consistent level of control, any size design team can produce an integrated design that is flexible and reusable. The VIPA Vehicle Design team utilizes a top-down, layout based Computer Aided Design (CAD) process to control every element and interface in the vehicle.

Top-down, layout based design is a method of designing complex systems consisting of a hierarchy of control model files, part files and assembly files utilizing CAD programs. Control files typically consist of a top layout file and subordinate layout files, and can be considered as a 3D interface control. The purpose of the top layout file is to determine the location, size and shape of the major components or subsystems of the design through the use of sketches, datum planes and axes, coordinate systems, and control surfaces such as Outer Mold Line (OML) surfaces. Subordinate layout files, which are initially populated with geometry features linked from the top layout file, are used to create lower level layout files and part files. Each part file, linked to its controlling subsystem layout file, usually consists of a single part. Subassembly and top assembly files, independent of the control files, are created and populated with the linked part files.

The layout structure exerts total control over the relationships between the parts. This allows design changes to be rapidly propagated through a large assembly by modifications of only a few expressions or sketches – rather than individually manipulating every part. Design reuse is encouraged because in the normal course of developing designs, a “library” of easily resizable component assemblies is created. In addition, top down design is conducive for a design team to work simultaneously on the different subsystems and parts while maintaining control over top-level design constraints.

This methodology of top-down design is made possible by the geometric linking of parts, and the general approach is available to any CAD system capable of linking geometry. The VIPA process is based on the Unigraphics NX “WAVE Systems Engineering” (Reference 3) methodology originally developed by Boeing Phantom Works (Reference 4) and incorporated into the NX suggested best practices. This method separates the control structure from the assembly, using the NX WAVE tools to link the required geometry between files. The method has worked well for a design team on the order of 10, but can easily be expanded to a team much larger than that, as well as being useful to the individual designer.

Figures 4 and 5 detail the relationship between the control structure and the assembly structure. There is typically a layer of control structure, i.e. layout files, to match each

layer of assembly structure, i.e. part files and assembly files. The top-level layout file contains geometry features, such as the vehicle OML surfaces, and expressions required to define the interfaces and common parameters of the 2nd level assembly components. Each level of layout files below that contains interface definitions and common parameters for the next level down. The geometry and expressions that make up the interface definitions are linked between layout files using the WAVE geometry linker and linked expressions.

The control structure and assembly structure come together at the piece parts. The lowest level layout is linked to the parts in that assembly using the “Create Linked Part” (CLP) functionality in NX. CLP creates a very robust link between the layout file and the part that is difficult to break. Once the parts are built, they are assembled in absolute coordinates in new assembly files. The tight relationship between the lowest level layout, its parts, and the associated assembly creates a portable unit. This unit can easily be copied, resized and reused by bringing it into the more loosely linked higher level control structure.

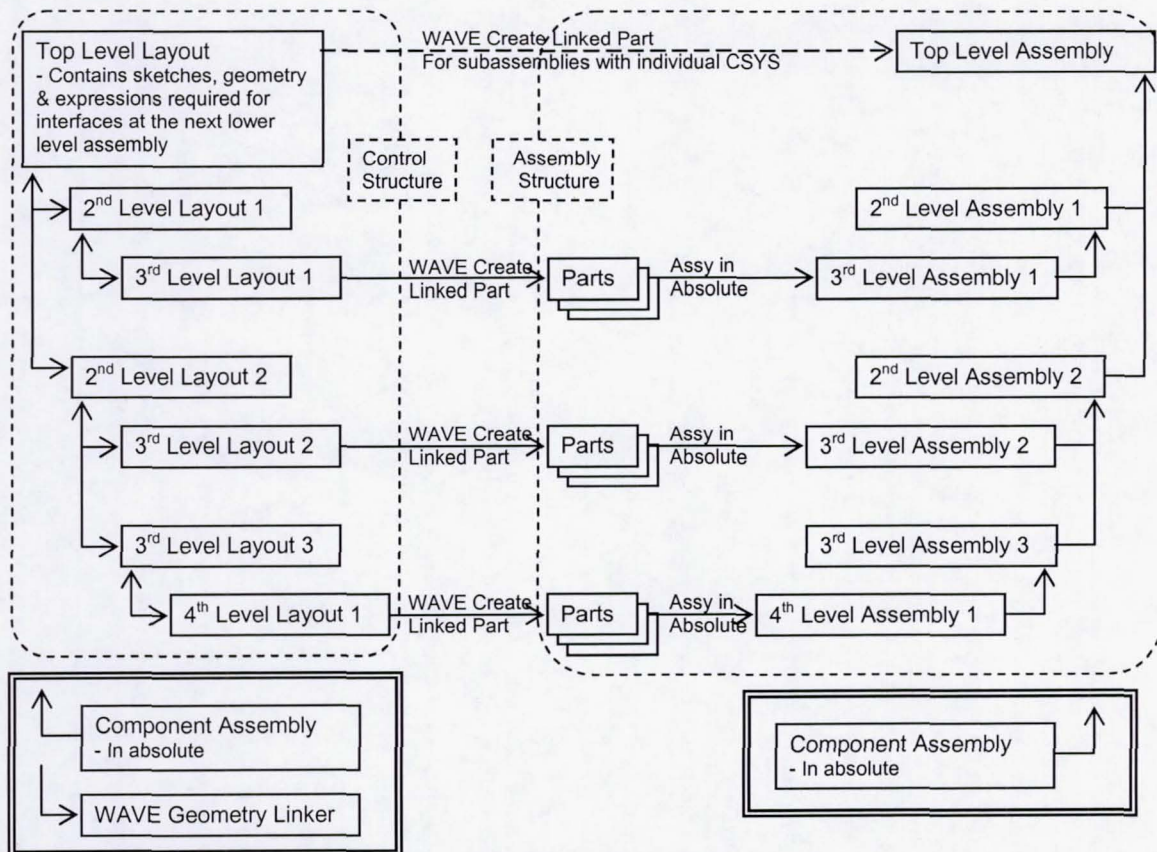


Figure 4: WAVE Systems Engineering Process Structure

The VIPA design process was developed during the early analysis cycles and has matured as the team gained experience and as additional capabilities were added to the NX program. Since the first design cycles focused on evaluating conceptual designs of complex winged vehicles for SLI, much of the early effort was focused on developing

robust models that could be morphed (or copied) to different locations to quickly populate the thousands of structural parts of the vehicles. Almost all of the parts interfaced with the complex outer mold line vehicle surfaces, so advanced surfacing techniques were utilized to allow the parts to be morphed reliably. This effort also allowed the entire vehicle design to be reused to create different vehicle stages with slightly different shapes.

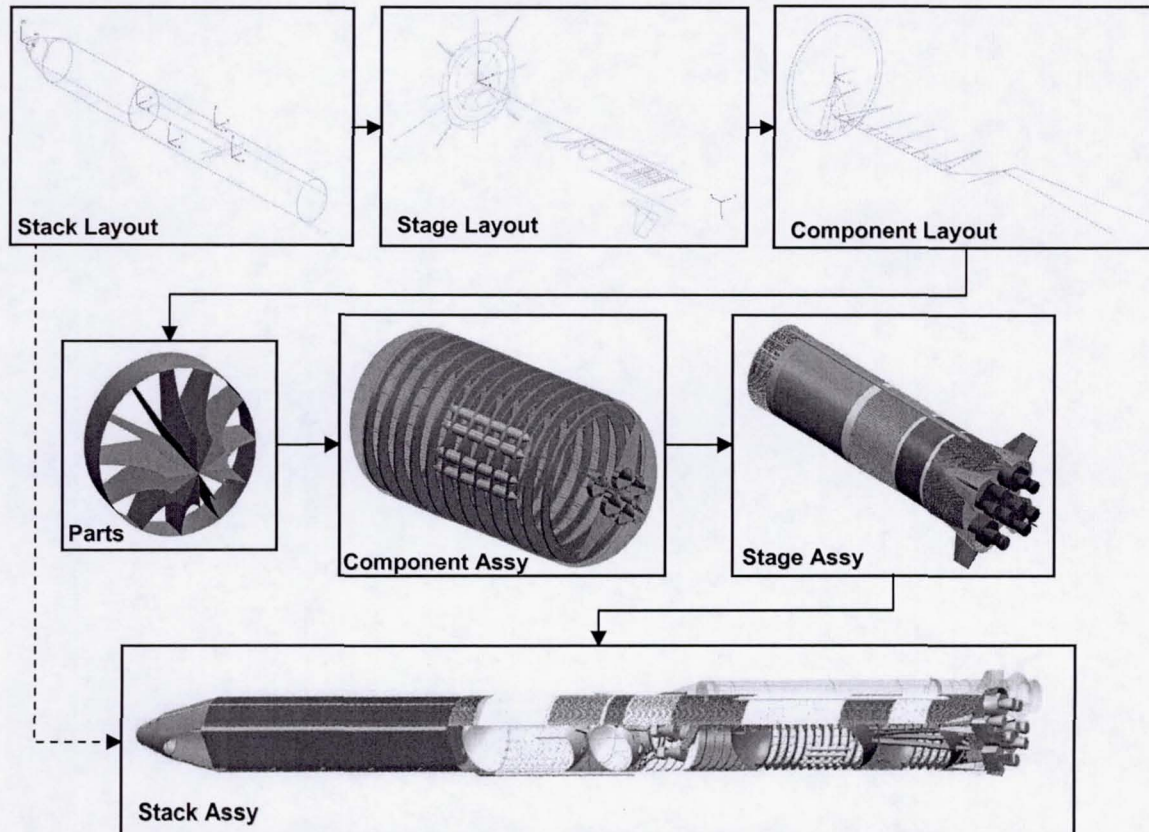


Figure 5: WAVE Systems Engineering Process Example

The Saturn V design cycle focused on an existing design, but the vehicle was large and needed to be modeled quickly. The relatively simple geometric shapes made it possible to create the large number of robust parts with comparatively few layout files. This robustness facilitated system trade studies, which resized the vehicle based on modern materials and improved engines.

The Orbital Space Plane design cycle presented some unique challenges due to the large number of booster and spacecraft combinations. While each vehicle segment maintained its own local coordinate system, the shapes of the interface segments (the payload adapters) were dependent on the relative positions of the booster and spacecraft. The relationships between the individual segment coordinate systems were developed at the top-level layout, and the interface outer mold line geometry was developed at that level as well. The coordinate systems were linked directly to the top assembly by CLP, with the segments assembled and mated to their specific coordinate system. After the initial design of the payload adapter and crew escape system was modeled, it could be quickly

cloned, or copied, and connected and resized for the new vehicle combinations, including the non-axisymmetric shape of the winged OSP concept.

The Shuttle-derived heavy lift booster design was a combination of existing and new components. The Solid Rocket Boosters and External Tank were not modified from their existing designs, while the orbiter's Space Shuttle Main Engines (SSME), thrust structure, and propellant feed lines were integrated into a brand new payload carrier design. The large payload carrier geometry required significant changes to the geometry of the aft section of the vehicle that houses the engines, feed lines, and thrust structure. After the initial configuration was analyzed, a derivative configuration was quickly developed which altered the structural arrangement for a different payload separation scheme.

The most extensive example of design reuse was the development of the clean sheet heavy-lift design. While the specifications for the vehicle were brand new, the first two stages of the Saturn V models were copied and modified to create the new design. The flexibility and robustness of the control structure allowed major changes in vehicle size to be easily propagated to the detailed part models without a significant amount of rework. Copying components of the oxidizer tank allowed quick alteration of the second stage fuel tank configuration. Alternate configurations of the vehicle (with and without Solid Rocket Boosters) were created using the Variant Product Structure capabilities of the NX integration to the PDM without having to duplicate the top-level assembly file. Roughly 6 total manweeks of effort were all that was required to completely resize and redesign a 4000+ part assembly.

Managed Development Environment

The Teamcenter Engineering (TCEng) PDM (Reference 5) is used to administer VIPA project data. This software package is a database containing all project data including Computer Aided Design (CAD) geometric models, CAD drawings, documents, specifications, analysis files and presentations. This system facilitates collaboration at multiple sites and ensures that all team members use the same up-to-date data and assumptions. It assists configuration management by controlling part checking and release. Project data is organized into Items and Datasets, as well as an informal, team developed folder structure. The Master Model approach is utilized to organize part data. The Variant Product Structures function allows assemblies to represent multiple configurations.

Items and Datasets

Items are the fundamental objects used in the TCEng database. Items are structures used to represent a part, layout, or assembly, and are typically equivalent to a part or document number (Figure 6). Each Item has a unique identifier, the Item ID, as well as a name and a description. A level under the Item is the ItemRevision. As an object changes over time, it can be saved as a new revision. Other part attributes such as mass and CAGE code are stored with each ItemRevision. Under each ItemRevision are datasets that are

associated specifically with that revision. For CAD files, there is always a dataset for the CAD Master. CAD Datasets consist of several files including the CAD part file itself, additional tailored part attributes, exported files, and external images that may be used in the file. Other datasets that can be in the ItemRevision are linked CAD files, translations for other CAD software systems, documents (engineering orders, mass properties reports, material specifications) and analysis models (such as finite element or motion analysis). A lightweight geometric model is generated from the CAD files in another dataset to allow 3-D models to be opened by team members without expensive CAD software or expertise.

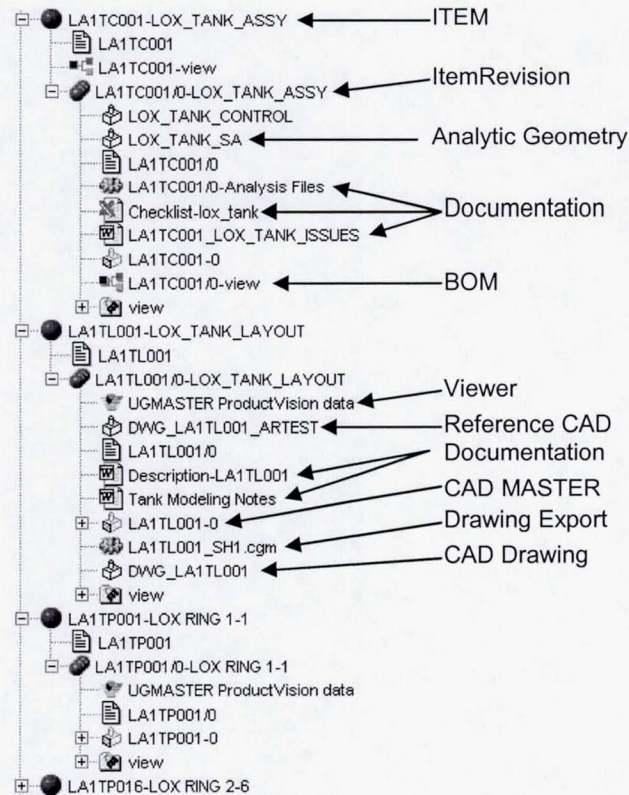


Figure 6: Items and Datasets in TCEng

Master Model Approach

The “Master Model” is a methodology for organizing CAD information by separating the product definition of a part from other part data. The “master” CAD file holds all of the data that is needed to fully define what the part is, but nothing more. Other CAD files hold data derived from the master. This can include individual datasets for drawing files, Numerically Controlled (NC) machining data, analysis models, and files that simplify the master data for specific uses. This approach facilitates concurrent engineering because the data for different disciplines is separated and can be developed simultaneously. Since this data is stored separately, the Master Model is smaller and will use less memory when fully loaded. Access controls can be applied separately so that, for example, the master and drawing are protected after the release cycle, while NC code can still be modified, and perhaps locked down at a later date. In effect, the non-master files are similar to a

CAD assembly whose only component is the master file (which may itself be an assembly). In TCEng, the non-master files are maintained as additional datasets within each ItemRevision. Based on the user's choice, non-master files may be duplicated when a new ItemRevision is saved.

Variant Product Structures

Variant Product Structures allow one assembly to represent different but related configurations. An automotive application of this capability would be car models. For example, a car could have a GL and an LX option package. Within those top packages several additional option packages and individual options are identified. Maintaining multiple separate assemblies would require more effort and introduce errors if configuration changes were not made in all assemblies.

The Variant Product Structures function is implemented in TCEng by adding conditional statements to the product structure. This technique was used in the analysis of different Clean Sheet Heavy Lift Vehicle Concepts because of the high degree of commonality between the Core Vehicle and the Core-SRB Vehicle. By setting a couple of top-level variables, the entire product structure of that specific vehicle can be generated.

In this case, the differences for going from the Core-SRB Vehicle to the Core Vehicle were the removal of the Solid Rocket Boosters (SRBs), the crossbeam, and SRB attach fittings, and the addition of the Stage 1 fins. Figure 7 shows the TCEng interface with the definition that when Launcher=TC-SRB Vehicle (nicknamed BamBam), the right SRB is to be included. The circled V next to the part number indicates that part is controlled by a variable rule.

Figure 8 shows the NX CAD view of the defined variable product structure. The Core Vehicle (Pebbles) has been selected, and the SRBs do not appear in the product structure, but the fins on Stage 1 do. This is an effective way to make sure that multiple users of an assembly are dealing with exactly the configuration they expect when generating derivative data for downstream applications.

Vehicle Design Team Interaction with the Rest of VIPA

There are some issues that make interaction within a systems analytical team challenging. Normally, the members of a particular discipline team are located closely together, but are separated from members of the other teams. This can complicate communication and the sharing of data, particularly of sensitive data. Members of the different teams are primarily skilled in the particular discipline of their team and often have little knowledge about the other discipline teams. Another challenge is that different discipline tools often do not interact smoothly. Lastly, the disciplines rely on outputs from each other, leading to a lag as teams wait for their input data to be generated. Many of these issues are not easily resolved. For instance, co-locating the members of all of the VIPA analysis teams is not realistic because of 1) a lack of dedicated, available space, 2) most VIPA members are working other projects at the same time, and 3) discipline software support is more efficiently provided when the users are physically close.

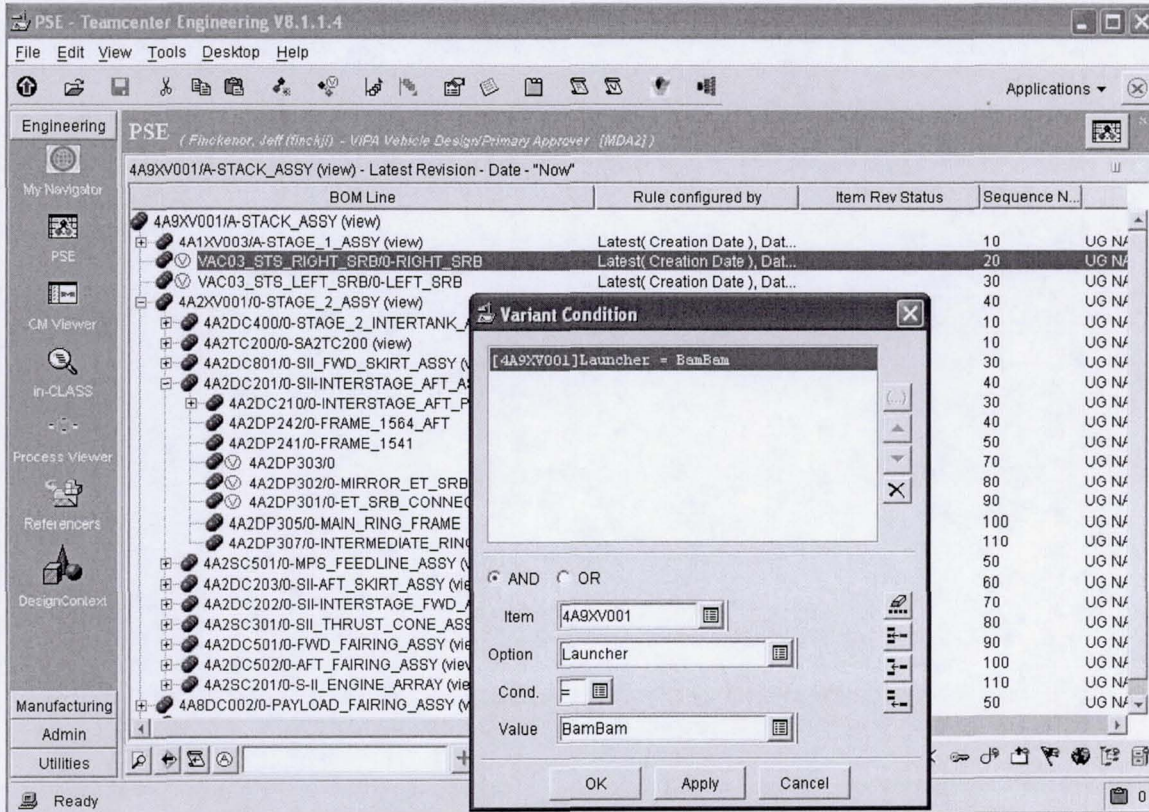


Figure 7: Variant Product Structure Teamcenter Engineering View

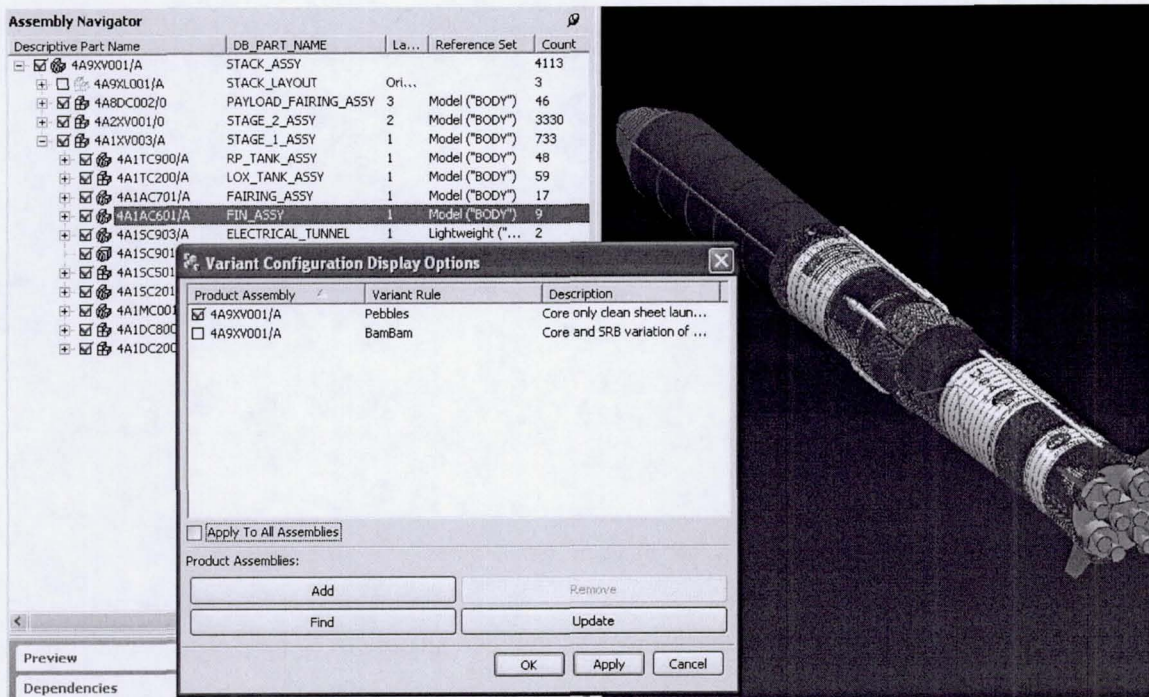


Figure 8: Variant Product Structure NX View

PDM and Access Control

To help address these issues, VIPA applied the Teamcenter Product Data Manager. Teamcenter provides a secure method of sharing model and data files. A folder structure was implemented real time by the team members to organize data into general use and discipline specific areas. Access control was arranged to allow each VIPA team to read, but not edit, other discipline team data to avoid inadvertent changes. In Teamcenter, this means each discipline team had its own "group", and access to data in the database is controlled by a rule tree based on what group the accessing user is logged in as. The user group can be easily changed on the fly if needed, or the owning group of a particular dataset can be easily changed once created. All members are in a group for their own discipline, as well as a general "VIPA Team" which can be used for group editing of systems documents.

Several interfaces are available to Teamcenter, each of which is most appropriate for a different function. CAD work is performed using the NX/Manager interface, which is a direct and seamless interface through the NX software. File operations using NX/Manager are nearly identical to native OS file operations except that they operate directly in the database rather than on local drives. The thick client, or Portal interface, can be used for all PDM functions including access control changes, product structure editing, etc. The thin client web interface is typically for non-CAD users to share files, as well as have access to the lightweight visualization models.

Concurrent and Systems Engineering

The VIPA Team structure, discussed above, was intentionally arranged to help with the lack of inter-disciplinary knowledge by the team members. Early in the formation of the VIPA Team, an NXN diagram (Reference 2) was used to help map out the data that was required between disciplines. An NXN is simply a matrix with each discipline along the diagonal; inputs to that discipline are along the columns, and outputs along the rows. It is a useful tool to help discipline engineers understand how their data fits into the system. By studying the needs of the other disciplines it is possible to find better ways to perform concurrent engineering.

For example, it is clear that Loads and Dynamics and Structural Assessment need geometric descriptions, however studying the NXN and striving to understand the other disciplines led to the realization that the top down modeling approach used by the Vehicle Design Team was particularly conducive to concurrent engineering. Early Loads & Dynamics finite element models tend to use beam elements. All the geometry needed for the beam model is in the Design Team layouts, which are the very first files to be generated. The early Structural Assessment finite element models are simple sheet body models, sometimes integrated with structural optimization codes that use shell models for element load definitions. The simple shell models can be generated quickly from the layouts, using a file linked to the master layout model in the managed environment. While the loads and stress models are being run, greater detail is being added to the

design models. When they are complete, accurate beam and shell properties can be easily updated in the other disciplines for a rapid completion of an analysis cycle.

Model Checking and Configuration Control

It is important during trades that everyone understands what configurations are current and up to date. To meet this need, the Teamcenter cascade release process was implemented for VIPA. It was used primarily in VAC-00 and VAC-01 since in the other cycles there was never more than a single configuration. Also, due to the time constraints, the fact that files were changing rapidly was understood by all. Several important workflows were developed to meet the needs of the team. The initial workflow is a "Vehicle Design Checking" process, which results in the models being locked down in the database. This is followed by a "VIPA Baseline" process, which is an indication from each of the discipline leads that they understand this is the configuration under current study. A final "VIPA Complete" process can be used to indicate that each discipline has completed working on a given configuration.

The Vehicle Design Checking process is a model review similar to the traditional drawing checking. Things being checked include part number, naming, and layer conventions, all attributes and reference data (including viewer data) exist, reference files are current, the state of the file on save is correct, and the product structure had been set to precise. The Vehicle Design team members themselves perform the checking, but always on files generated by another member of the team. Many aspects of the checking were automated, and designers are encouraged to run the script on their own parts prior to submitting them to a workflow.

Lightweight Viewer

To help communicate CAD and 3D information, Teamcenter is integrated with a product called VisView (Reference 6). Each CAD package managed within Teamcenter (most notably for MSFC: NX, Catia, ProE and Solid Edge) writes out a lightweight 3D format. VisView is integrated with the web interface providing an easy method for non-CAD users to access the 3D geometry (Figure 9). This allows all users a real time ability to visualize and measure the system being studied. In addition to viewing CAD models, VisView allows the viewer a wide range of activities from taking measurements to creating animations of the models, with no danger of permanently changing the original models.

Particularly valuable to the productivity of the designer is the immediate access of the design information to the whole team. Traditionally, designers spend an inordinate amount of time printing draft drawings, copying them, and getting them delivered. Since the designer's data is now immediately available through the PDM, these non-value-added activities are no longer needed. There is a culture change involved as well. Newcomers to the VIPA team often ask "can't you just print it out and fax it to me?" The designer must answer "no" in order to maintain team efficiency.

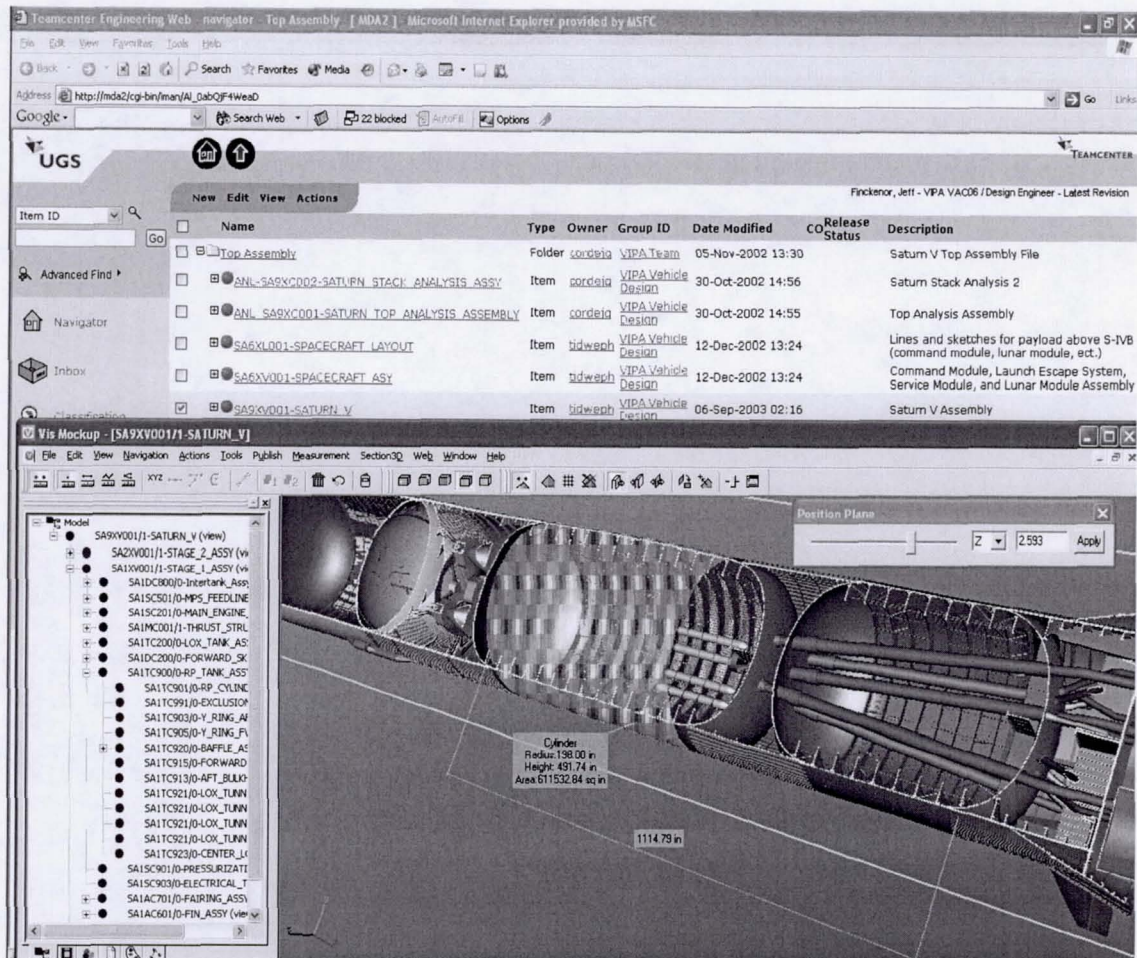


Figure 9: Viewer Launched from Web Interface

Application of the Master Model Concept

Simplifications or extractions of the master CAD models are needed by many of the VIPA Discipline teams. The design team has taken the responsibility of providing that data. Model simplification has long been the challenge in transferring detailed design models into analytical models. It can be argued that analysts are best suited to this task because they know exactly what they need the end result to be. VIPA's approach takes advantage of the skill of the designer, who is most practiced at manipulating geometry and has the best tools for performing that work (many analytical geometry interfaces are primitive compared to modern CAD). Also, CAD models generated by VIPA designers are in the managed development environment and are controlled, via the Master Model approach, by the current models. VIPA designers, by working with the other analytical teams to understand what their data is being used for, are able to provide precisely what is needed for each discipline. The goal of VIPA Vehicle Design is to provide mesh ready geometry for each analysis.

The Vehicle Design Team interacts with each of the other VIPA discipline teams in unique ways, with some of the products outlined in Figure 10.

- Loads & Dynamics receives layout sketches, line models and Outer Mold Line data early in the cycle. Later in the cycle, cross sectional properties of assemblies can be provided in tabular format. Vehicle Design may directly receive stiffness requirements for certain members, as well as loads received indirectly as part of the integration with stress.
- Structural Assessment receives sheet body models and can be involved in tight iterations with Design to obtain optimum structures.
- Thermal receives OML models and occasionally curvature measurements at specific points. From Thermal, Vehicle Design receives TPS material information and design thicknesses, as well as purge line sizing when needed.
- Flight Sciences (Aerothermal & CFD) receives OML surface models for Computational Fluid Dynamics analyses. When not using CFD, the layout information is used for generating aero via traditional methods. Vehicle Design has received engine plume shapes to aid in the positioning of booster strap-ons to avoid impingement.
- Propulsion receives tables of line locations, angles and lengths, as well as engine position data and gimbal points. Vehicle Design receives geometric engine data to allow for packaging and line routing.
- System Modeling maintains the system mass properties and performs visualization and simulation studies. A major product of Vehicle Design is the nominal mass properties of the system, which the Systems Modeling team then manipulates and provides to the rest of the VIPA teams. Visualization models are generally sheet bodies of the OML, sometimes with additional internal geometry. Vehicle Design receives mass data for elements that are defined elsewhere so they can be incorporated and rolled up in the 6 degree-of-freedom mass properties.
- Avionics/Power receives antenna and box locations. Vehicle design receives equipment lists, including size and mass data, for packaging into the system.
- Flight Mechanics, via the Systems modeling team, uses the mass properties generated from the Vehicle Design models, as well as thruster locations and vectors. Feedback to Vehicle Design is indirect through the environments and loading studied by the other disciplines.
- Material & Manufacturing receives definition of the piece parts and system, and can provide detailed manufacturing plans for major components.

An excellent example of the value of concurrent engineering using the master model occurred during VAC-03. The initial concept was developed and laid out following the VIPA team practices. While geometric detail was being added the Loads & Dynamics team determined that a primary structural beam could not reasonably be made stiff enough to meet frequency constraints. Within hours of finding the problem, the team had discussed the options and determined the most reasonable course of action. The detailed geometric design was altered, and the trajectory and simulation data was made to reflect the change in flight plan. Part of VAC-05 was trading fairing attach and jettison schemes involving this change.

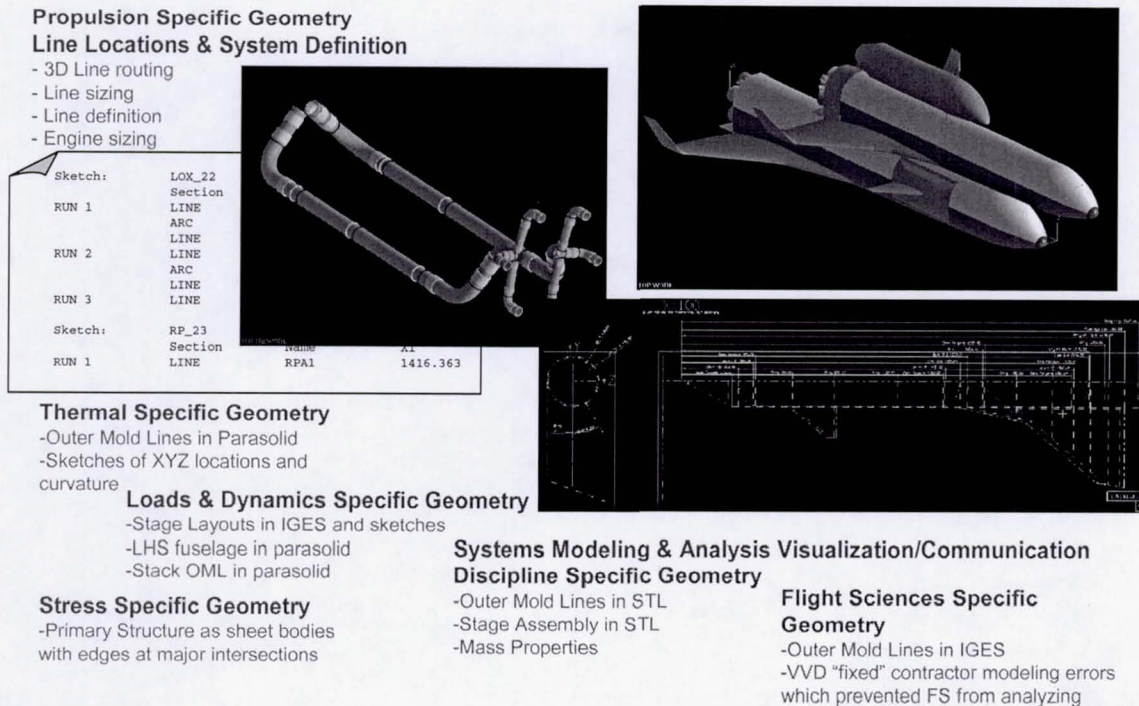


Figure 10: Vehicle Design Team Interactions in VAC-00

Mass Properties

One of the most important deliverables of the Vehicle Design team is a detailed set of nominal mass properties. A complete spreadsheet of part and assembly mass properties, in the form of an indented parts list, is delivered to the Systems Modeling team. The Systems Modeling team then applies margins as specified by the project, adds elements which were not modeled in detail (such as the Solid Rocket Boosters), and develops sequenced mass properties including propellant expenditures and jettison masses. This formal mass property report feeds the 6DOF flight simulation, which in turn allows induced environments and performance analyses. The need to generate meaningful mass properties for the vehicle is the primary driver for the level of modeling detail employed by the Vehicle Design team. In the case of modeling structural elements, feedlines, TPS, etc., it is best to model the part to an adequate level of detail and assign the appropriate density.

In a complex vehicle system, it is also common to have mechanism or other components that are 'off-the-shelf' items. These types of components work best with an assigned mass since they will probably be modeled with very little detail. Since these items are typically small in relation to the whole system, assuming a uniform distribution of the mass within the volume has a negligible effect on the vehicle moments and CG location. The method for doing this was to divide the target mass by the volume of the model and manually set the density. To simplify this task, a Set Mass Value macro was created to automate the function. The macro prompts the user to select a solid then prompts for the desired mass value.

Once masses and/or densities have been correctly assigned to all components, mass properties can be generated for the whole assembly. The Assembly Weight Management function in NX provides the capability to send this data directly to an Excel spreadsheet. This function has options that allow the user to choose which components and which reference sets should be included in the mass properties calculations. An Excel macro was written to insert columns for Part Revision and Part Name for each component from the database. Use of the Assembly Weight Management function makes generation of highly detailed mass data extremely easy for the user. It is also a valuable model-checking tool, as parts with incorrect densities or unreasonable weights can be easily spotted and fixed.

Special CAD Macros

In the course of modeling work, the Vehicle Design team would find certain functions or activities that could be tedious to perform. In these cases, scripts were developed within the NX environment to help increase the designer's productivity, such as the Set Mass Value routine discussed in the Mass Properties section.

Mirror Components

When creating assemblies of symmetric vehicles, it is common practice to model only the left-hand parts where possible. To accommodate this idea the VVD team uses a part numbering convention that assigns odd numbers to all explicitly modeled parts and reserves the even numbers for right-hand parts (mirror images across the plane of symmetry). Each mirror component would have the next number in sequence from its counterpart.

The Mirror Component macro was created to perform this function automatically from inside the assembly file, since the manual method is somewhat labor intensive. The macro creates a new component with WAVE-linked Mirror Bodies for all the solids in the NX BODY reference set of the left-hand part. While newer versions of Unigraphics NX have the Mirror Assemblies Wizard, which can perform the same function as this macro, there is no option to create a part name using the numbering rule above. Because of its power in populating a large design, simplicity, and compatibility with the VIPA numbering convention, the VVD team still uses the Mirror Component macro for right-hand part creation.

Other Useful Macros

While CAD users at MSFC have a variety of macros to help perform their functions, those which were developed for VIPA are summarized in Table 1.

Table 1: VIPA Developed Macros

NAME	FUNCTION	PURPOSE
Set Mass Value	Calculates and assigns density to make an identified solid a specified mass	Off-the-shelf or otherwise defined bodies that are not modeled in detail need to have known masses applied for proper roll up in a mass properties report.
Mirror Component	Creates and adds mirrored part files within an assembly	Many VIPA cycles have involved systems with at least a plane of symmetry, allowing one side to be modeled and then mirrored to generate the whole assembly.
Calculate Section Properties	Calculates mass properties at section plane locations	Loads and Dynamics beam models require accurate section properties for analysis. Output from this macro is suitable for direct input to the Loads and Dynamics finite element model.
Set Body Type Preference	Quickly toggle between Sheet Body and Solid Body creation	Robust modeling techniques may require several switches between solid and sheet body creation.
Calculate Flow Data	Calculates flow data for fluid lines based on path curves	Propellant flow data is required for engine performance analysis.
Reset Name Location	Resets name display to appear centered on object	Names do not move with objects when design changes reposition them. Moving names manually is tedious and time consuming.
Reset Sketch Dimension Location	Moves sketch dimension origins close to their associated objects	Dimension origins do not automatically move with sketch objects when the sketch position changes.
Setup BODY Reference Set	Creates or updates BODY reference set for any file type	Automatically sets the NX BODY reference set to meet the VIPA file standards.
VIPA Check	Performs several checks to ensure parts meet VIPA standards	Macro assures files are created to VIPA modeling standards and prevents many common errors.
Update File Data	Sends file information to the Teamcenter database	Macro provides a quick method to export images, weights, solid geometry, and drawing data to TeamCenter for easy reference by other VIPA teams.

Summary

Vehicle Design Practices

VIPA top down design, as modified from the NX WAVE Systems Engineering, is used as a highly effective method for controlling and manipulating the definition of large, complex systems. It allows for rapid design development, and enables and encourages effective design reuse.

The Master Model approach, especially applied within the Teamcenter Engineering Managed Development Environment, has proven especially well suited to concurrent engineering and effective interaction with a wide variety of engineering disciplines.

A minimal amount of development effort, applied to aids for very specific design functions, can greatly enhance the productivity of the designer. Both design specific productivity, as well as enhanced interaction with other disciplines has been improved.

The use of 3D visualization tools within the Managed Environment is another major asset to the productivity of the designer, because it removes the tedious tasks of generating and distributing paper renditions of his work.

The successes of the Vehicle Design Team have given a dramatic demonstration of the value of its practices.

The VIPA Team and Analytical Integration

The VIPA Team has demonstrated a unique ability to generate a significant amount of detailed, integrated information in exceedingly short periods of time. VIPA has been able to aid program decisions by providing critical information very early in the development process. VIPA emphasizes process and interaction among the members of the team, while relying on the familiar and validated tools of the experts. With this approach, VIPA is not limited to any specific type of system as many tool-centric approaches have been. To date it has been primarily applied to launch vehicles but any system could be approached in the same way.

The general model is a focus of the VIPA development effort, but the resulting tool itself is less critical than the interaction required developing it. The general model acts as a focus of communication and discipline interactions. It is a method for expanding the scope of the discipline engineers to the overall system.

NASA projects need detailed information as early as possible in order to plan and manage effectively. The history of NASA projects has shown repeatedly that complex systems require effective government insight. Complex systems require independent reviews to improve the probability of success. VIPA has dramatically demonstrated its ability to be a valuable asset to Projects by providing rapid and detailed technical insight early in the development, for either new or contractor delivered designs.

References

- 1) McGhee, David S., "Vehicle Integrated Performance Analysis, The VIPA Experience – Reconnecting with Technical Integration", NASA-TP-(pending), 2005
- 2) Blair, J.C., Ryan, R.S., Schutzenhofer, L.A., Humphries, W.R., "Launch Vehicle Design Process: Characterization, Technical Integration, and Lessons Learned", NASA-TP-2001-210992, May 2001
- 3) NX 3.0.1.3, Online Documentation, UGS Corp. 2004
- 4) Phillips, J.R., Frey, E.K., "Three-Dimensional Solid Modeling in Aircraft Design", AIAA-99-1364, 1998
- 5) Teamcenter Engineering, V9.1.2.0, UGS Corp. 2004
- 6) Teamcenter Visualization, VisMockup 5.1, UGS Corp 2004